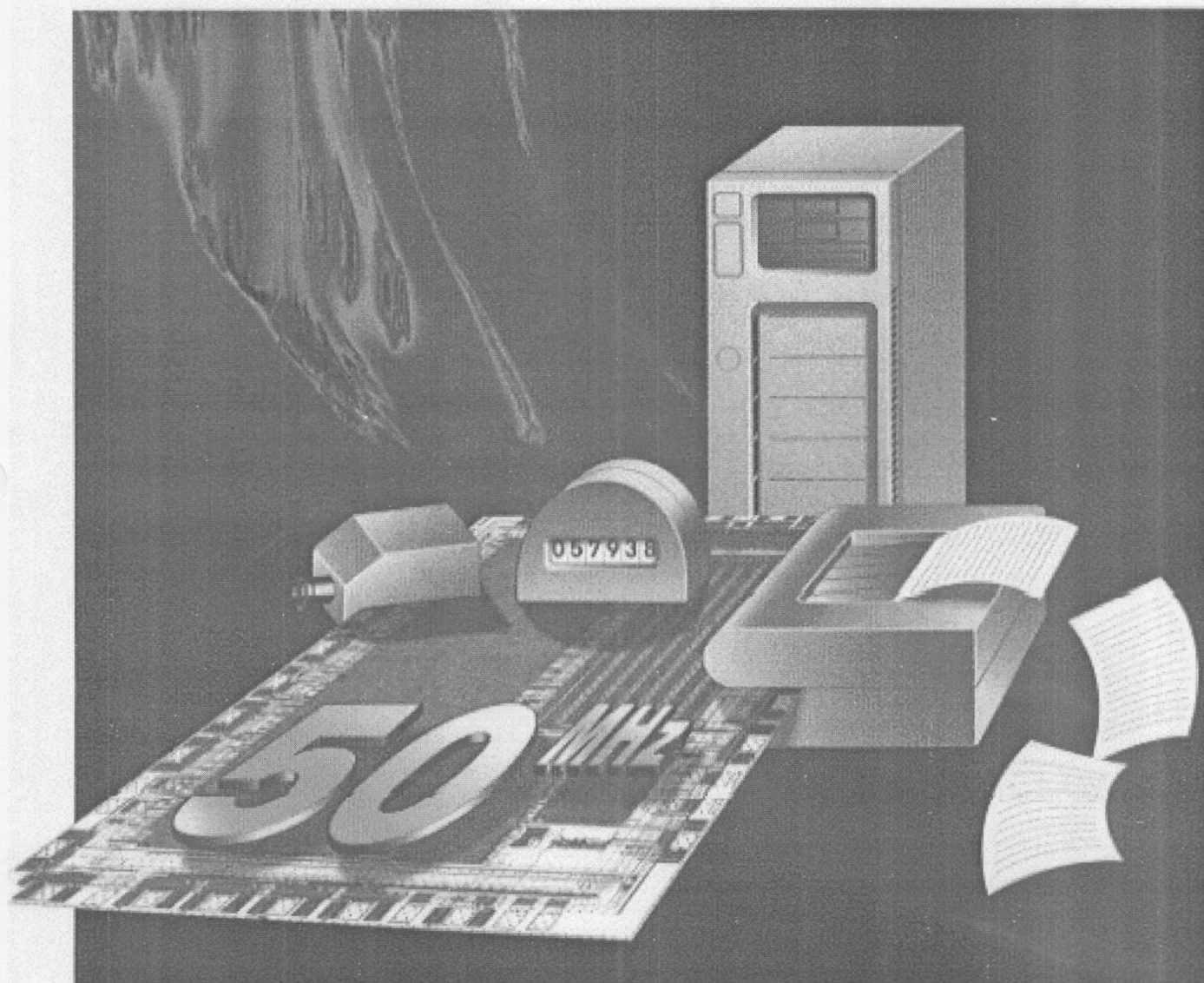


# ELECTRONIC DESIGN

FOR ENGINEERS AND ENGINEERING MANAGERS - WORLDWIDE

A PENTON PUBLICATION \$10.00

AUGUST 4, 1997



## **Speedy 8-Bit Microcontroller Crafts Virtual Peripherals p. 36**

**802.11 Wireless LANs: A Blueprint For The Future p. 44**

**The Universal Serial Bus Is Ready To Take Off p. 56**

**ISLPED'97 Shows What's Up In Ultra-Low-Power Digital Systems p. 65**

**Board-Level Connectors For High-Speed Systems p. 75**

**The IEEE P1451.2: A Smart Transducer Interface For Sensors And Actuators p. 97**

**CEE Approval Means Understanding The EEC EMC Directive p. 108**

# TECH INSIGHTS

■ Exploring the world of digital, logic, memory, and microprocessors

## Speedy 8-Bit Microcontroller Crafts Virtual Peripherals

*With A 50-MIPS Throughput, This MCU Uses Software To Create Application-Specific Virtualized Peripheral Support Functions*

ave rs

Single chip 8-bit microcontrollers solve many embedded control applications by inexpensively integrating many system control functions along with the processor onto a single chip. However, as system needs grow more complex, the silicon area of the microcontroller unit (MCU) also grows—often to the point of making the chip too expensive for the application. Furthermore, many of the application subtasks (counting, timing, serial and parallel communications, pulse-width modulation, etc.) do not demand the ultimate in CPU performance, but can usually be accomplished with a CPU that runs at a few megahertz, since on-chip logic supports those functions.

To lower the cost of the silicon, Scenix Semiconductor has a better idea. Instead of putting application support functions in hard logic, they use software to create virtual peripherals. Each support function is actually a software module that can be incorporated into the MCU's program storage memory. Modules are executed from memory when the functions they represent are needed. Thus, each function requires only a portion of the CPU's overall throughput, limiting the number of support functions a CPU can handle simultaneously. However, by using software, custom silicon development is eliminated in many



time multiplexing provides the appearance that they all simultaneously exist (see the table). However, sometimes additional hardware must be used due to resolution or speed

requirements. High-resolution ADCs

(beyond 10 bits), for example, are best implemented as separate chips because their integration with the MCU usually brings conflicting process requirements.

### Overhead And Code

Just how much over-head and/or code do the virtual peripherals require? Not too much for some specific functions, and most of the CPU's capability for others.

For example, code needed for an IR transceiver interface is about 170 words of EEPROM. This code requires about 5% of the 50-MIPS throughput, leaving about 45 MIPS for other tasks. An SPI/Microwire serial port takes about 50 words of storage and about 10% of the CPU throughput. A stepper-motor controller also requires about 50 words of storage, yet demands only about 2% of the CPU throughput.

Popular functions such as a keypad scanner or LCD controller require only 70 or 120 words of code, respectively, but both consume just 1% of the CPU

cases. As a result, silicon costs and hardware turnaround time drop.

After analyzing support-function requirements, Scenix decided to develop the fastest CMOS 8-bit microcontroller—the SX. The chip can run at a clock rate up to 50 MHz. With its one-instruction/clock execution rate, it can deliver a throughput of about 50 MIPS. The chip is designed to operate from supply voltages as low as 3.7 V, consuming just 12 mA at 50 MHz. The circuit, however, can operate over a supply range spanning the range of 3.7 to 6.25 V.

With such a high instruction throughput, multiple support functions might require from several thousand instructions/s to several millions of instructions/s to implement, can execute on a time-sliced basis. The

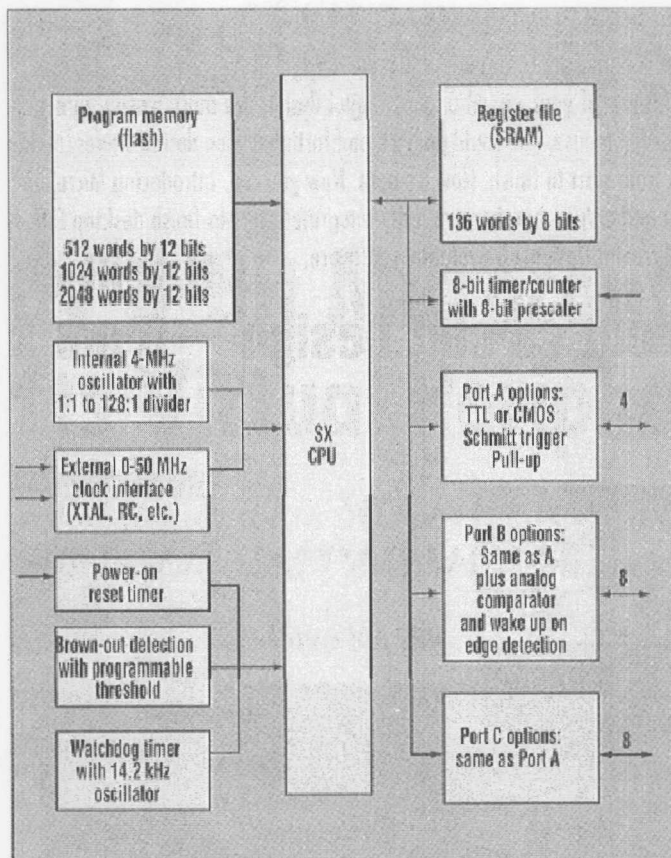


throughput. A full PC keyboard and mouse controller also would require about 80 words of code storage, yet the larger keyboard demands no more processing—just 1% of the CPU's throughput. More complex functions such as a 1200-baud modem or a music synthesizer require 280 and 180 words of code, respectively, but each consume 20% of the CPU's horsepower, leaving 40 MIPS for other tasks.

Since these and many other support functions can be implemented via software, the processor only needs to pack a minimal complement of on-chip support logic (Fig. 1). The wide operating voltage range, coupled with the on-chip brown-out detector, reset circuit, and watchdog timer eliminate the need for external circuitry such as voltage regulators and power monitor ICs. This design allows the MCU to run directly from batteries.

Key to the processor's flexibility is the high-speed 8-bit CPU core. It runs at 50 MHz, and has three programmable I/O ports (one 4-bit and two 8-bit ports) that can be used to implement virtual peripherals. The 4-bit port (RA) and one of the 8-bit ports (RC) provide TTL or CMOS interface levels with Schmitt-trigger inputs and pull-ups on the outputs. The other 8-bit port (RB) has the same features, but also includes an analog comparator and a wake up capability on edge detection.

The minimal feature set will keep the chip cost low. They also will allow it to serve as a low-cost addition to an embedded PC or RISC system. In this way, the chip would provide the I/O functions that don't make economic sense to integrate into the host CPU, or would add too many CPU cycles.



1. On the SX CPU from Scenix Semiconductor, most peripheral functions can be created with software and a minimal complement of hardware support. In addition to the flash program memory and 136 bytes of SRAM, the controller contains a 4-MHz clock oscillator, power-on reset timer, brown-out detection with programmable threshold, watchdog timer, 8-bit timer/counter with 8-bit prescaler, and three I/O ports—one with four lines and two with eight lines each

By using the SX to handle the I/O, the host CPU is free to handle more complex operations that better use the 16-, 32-, or 64-bit CPU data path.

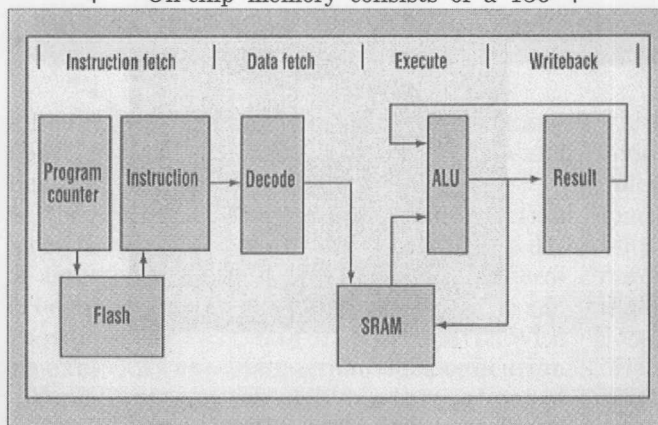
On-chip memory consists of a 136-

byte static-RAM-based register file, and a choice of either 512, 1024, or 2048 12-bit words of nonvolatile, flash-based program memory. The decision to use flash memory was critical in light of the performance objective—memory access times of just 12 ns, 3.7-V operation, and a 10,000-write-cycle endurance. The EEPROM also is in-system programmable. Consequently, a separate external programmer is not required for updates.

To implement the high-speed flash, designers at Scenix developed a new, asynchronous flash-EEPROM structure. Address transition detection also helps lower the memory power by deselecting the EEPROM between cycles. Once the address stabilizes, a differential sense amplifier with a bias in the linear region is turned on and fully saturated. Consequently, the flash EEPROM is enabled only 60% of the time. Furthermore, the EEPROM has a high signal-to-noise ratio with reduced loading and a reduced voltage swing.

The high-speed CPU structure

on the chip is similar to the Microchip PIC architecture, the first 8-bit microprocessor to employ a RISC-like pipelined architecture (Fig. 2). And the SX instruction set is an object-code-compatible superset of the Microchip PIC 16C5x family instructions. However, at the assembly mnemonic level, the SX also supports an opcode format previously defined by Parallax Inc. Designers can select whichever they prefer since it's a simple matter



2. A four-stage pipeline is used for the 8-bit CPU core in the SX microcontroller. The simple pipeline includes an instruction-fetch stage, data-fetch stage, execute stage, and write-back stage. This arrangement allows the CPU to achieve a throughput approaching 50 MIPS when clocked at 50 MHz.

to perform the 1-for-1 translation between the mnemonics

### In The Pipeline

To keep the pipeline simple and fast, there's no fancy interlock hardware, nor software delay slots that waste code space. In the pipeline are the instruction-fetch-and-decode-stage, the operand-fetch stage, the instruction-execute stage, and the write-back stage. Once the

pipeline is full, a complete instruction can be executed every clock cycle.

By relying on a streamlined four-stage pipeline with a Harvard-style memory structure, and employing extremely fast (12 ns  $t_{ACC}$ ) on-chip instruction and data memories, the SX can run at 50 MHz and execute almost any instruction in a single clock cycle. One exception, though, are Branch operations. They require three clock cycles and simply annul (replace with a

no-op) the two instructions that follow the Branch command. With a 20-ns instruction cycle, the chip packs enough punch to deliver about 10 times the performance of popular 8-bit MCUs from other suppliers. And, even when running with the low-speed 4-MHz internal oscillator, the SX delivers a higher throughput than a 40-MHz 8051-type MCU.

Although the MCU can run at 50 MHz, it doesn't have to, thanks to the use of static logic in the MCU's circuits. The chip can run at any frequency below 50 MHz (even stopped) without data or state loss. With the lower frequency comes the associated reduction in power consumption due to the CMOS logic. Designers can take advantage of the on-chip 4-MHz oscillator, for instance, since it offers eight programmable division ratios from 1:1 to 128:1. The on-chip oscillator reduces system cost, board space, and electromagnetic interference.

The 12-bit-instruction word width is also very efficient—it allows both the opcode and the operand in the same instruction. Additionally, the partitioning of the word between the opcode and operand is flexible, permitting maximum code density. And, to further optimize access to all the I/O features, 11 special-purpose instructions were added to speed common operations and reduce code size. For example, an additional carry-flag mode that works in conjunction with a carry-bit in the CPU significantly accelerates multibyte numeric calculations. It also yields performance approaching that of MCUs that have costly hardware multipliers.

Additional enhancements include a Bank instruction that switches the CPU to a new instruction page. Other new instructions deal with I/O mode bits and returns from interrupts.

### Speedy Interrupts

The processor's high performance also gives a very short response time to interrupts. The interrupts can be generated

### VIRTUAL PERIPHERAL CODE AND PERFORMANCE NEEDS

Peripheral function	EEPROM words	Percentage of 50 MIPS	Remaining MIPS
Dc-100-kHz frequency generator	30	4%	48
Dc-100-kHz frequency measurement	40	5%	47.5
100-kHz background timer	30	2%	49
I <sup>2</sup> C master	50	6%	47
I <sup>2</sup> C slave	80	3%	48.5
Keypad scanner	70	1%	49.5
PC keyboard and mouse controller	80	1%	40.5
IR transmitter	60	2%	49
IR receiver	110	3%	48.5
LCD module	120	1%	49.5
Dc-100-kHz resonance loop	130	10%	45
19.2-kbit/s serial UART	60	3%	48.5
Centronics parallel port	30	1%	49.5
500-kHz SPI/Microwire	50	10%	45
Dallas microwire serial port	130	3%	48.5
DMX-512	100	8%	46
X-10	60	1%	49.5
Video controller	130	20%	40
DRAM server	40	3%	48.5
Switch-mode power-supply controller	30	4%	48
Networking	130	5%	47.5
Music synthesis	180	20%	40
20-kHz 8-bit PWM output	30	10%	45
D-Σ ADC (1 kHz/8 bit; 4 Hz/16 bit)	50	10%	45
Dc-100-kHz spectrum analyzer	150	7%	46.5
DTMF I/O	120	10%	45
Caller ID	180	10%	45
300/1200-baud modem	280	15%	42.5
Speech synthesis	300	20%	40
Stepper-motor sequencer	50	2%	49
250-kHz quadrature encoder	40	10%	45
Servo loop	170	10%	45
PID loop	160	10%	45



by a built-in timer or external-pin edge detection. By exploiting deepened and dedicated hardware stacks, the SX guarantees response times of just 3 cycles (60 ns) for internal timer interrupts and five cycles (two extra for synchronization) for external inputs (100 ns total). One-level special hardware stacks automatically store critical registers during an interrupt and restore them upon return. That dedicated support lets MCU quickly switch tasks to implement virtual peripherals or execute application code specific to service that interrupt, all while executing code for other tasks.

Most older MCUs only interrupt tasks at instruction boundaries. In RISC processors, this limitation can be somewhat unpredictable because instruction lengths and the number of clock cycles required for instruction change depending on the instruction. As a result, interrupt responses are slower than those with the SX RISC processor. The slow response also may introduce jitter into system timing, which ends up forming a fundamental limit on system performance and accuracy. In contrast, the SX's response to interrupts is much more predictable and repeatable.

### Versatile Interfacing

The chip's I/O lines also incorporate the latest interface technologies, giving designers a flexible set of lines. All I/O pins are individually programmable as inputs or outputs, with or without 20-k $\Omega$  pull up resistors. Inputs offer selectable TTL or CMOS levels with optional Schmitt triggers. Every output can sink or source up to 30mA. The four pins on the 4-bit port (RA0-3) include symmetrical drive characteristics that permit the SX to easily drive inductive loads, which require equal push and pull (speakers, ultrasonic transducers, etc.). Additionally, three lines on the 8-bit RB port (RB0-2) can optionally connect an internal analog comparator to external signals. Furthermore, eight of the pins can be configured as external interrupt inputs. Besides redirecting program flow in response to external events, these inputs can wake up the chip from a

power down mode.

### On-Board Emulation

To ensure the SX controller is properly configured, built-in, in-circuit emulation circuits allow designers to use the standard SX chip to debug software and hardware operations, as well as program the on-chip flash memory. No special bond-out chip or complex tools are required, thereby lowering development cost. The approach combines the best of traditional ROM monitor and newer remote bug schemes, and is tailored to the SX's specific characteristics.

Since the SX is flash-based, it overcomes the single-chip-only dilemma—the application program can be repeatedly modified and downloaded into the chip along with a tiny debug kernel. To minimize memory consumption, the kernel handles only the bare minimum of functions needed to control the MCU (read and write instruction memory, go, and stop). All higher-level debug and human interfaces are performed on a host PC. The CPU's static nature also allow for true hardware single stepping or slowing of execution speed.

The low-pin-count packages planned for the SX (18, 20, or 28 leads) also limit the number of pins that can be dedicated to the background debug or a JTAG-like test port. To overcome the pin limitation, programming and debug operations take place over two clock pins (OSC1 and OSC2). This design works because these pins usually serve no application function other than clocking the CPU. That avoids any conflicts between the active I/O lines and debug signals.

To ease software development, Scenix created the SX development system. It includes an assembler, an in-circuit development tool that also doubles as a programmer and debugger, a demo board, and two sample parts. The kit is available from Parallax Inc., Rocklin, Calif., (916) 624-8333, and sells for \$249. Parallax also is developing a Basic compiler for the SX that will include an assortment of pre-designed virtual peripherals. The compiler and peripheral library will be available as an upgrade in the fourth quarter for \$99. Byte Craft Ltd. also is developing a C compiler and assembler for the SX. The company can be reached at (519) 888-6911.

### PRICE AND AVAILABILITY

The SX microcontroller with 2 kbytes of EEPROM sells for \$2.98 apiece in lots of 10,000 units. Samples will be available in September and production quantities will be available in the late fourth quarter.

Scenix Semiconductor Inc., 3140 De La Cruz Blvd., Ste.200, Santa Clara, CA 95054; Steve Leung, (408) 327-8888.